

Estudio 2



Estudio de la Arquitectura de la
Plataforma de Telemedicina



Índice

Sistemas Operativos.....	2
Lenguajes de desarrollo	4
Seguridad en el modelo cliente-servidor	8
Gestores de Bases de Datos	10
Diseño de la Arquitectura	12
Conclusiones	15
Referencias.....	16

Control de Versiones			
Revisión	Autor	Fecha	Motivo
1	José Criado SICBRAIN EUROPA SL	27/10/2011	Primera versión del Estudio 2.

Estudio 2. Estudio de la Arquitectura de la Plataforma de Telemedicina.

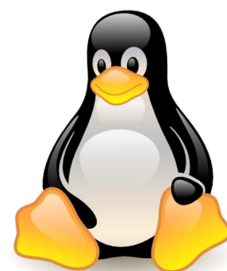
El objetivo principal de este estudio se basa en la definición y comparación de las diferentes tecnologías existentes para aplicar en la Plataforma de Telemedicina del consorcio RESATER. Para ello se realizará un análisis y la posterior defensa de los criterios aplicados, estando siempre en consonancia con las peticiones formuladas en el Pliego de Condiciones de la Licitación de la Plataforma.

La arquitectura de esta plataforma tiene que permitir dotarla de una serie de características tales como seguridad, robustez y fiabilidad. Todo esto a su vez estará expuesto en un entorno dinámico y sencillo para el profesional, siendo las capas inferiores a la interfaz de usuario las responsables de que el funcionamiento de la plataforma sea el correcto en todo momento.

Sistemas Operativos

Como Sistema Operativo (SO) la plataforma utilizará **Linux** en el servidor. Esto es debido a una gran cantidad de propiedades que hacen de este sistema operativo el idóneo para soportar una plataforma de estas características:

1. Multiplataforma: Trabaja en muchas CPUs distintas, no sólo Intel.
2. Multiusuario: Permite varios usuarios en la misma máquina al mismo tiempo.
3. Multitarea: Permite tener varios programas (realmente procesos) ejecutándose al mismo tiempo.
4. Linux funciona en modo protegido. Este es un modo operacional de las CPUs que permiten mejorar las multitareas y la estabilidad del sistema, así como protección de memoria y soporte de hardware para memoria virtual.
5. Posee protección de la memoria entre procesos, permitiendo que uno de ellos no pueda colgar el sistema.



6. Utiliza una política de copia en escritura para la compartición de páginas entre ejecutables; esto permite que varios procesos puedan utilizar la misma zona de memoria para ejecutarse, lo que aumenta la velocidad y reduce el uso de memoria.
7. La memoria virtual utiliza paginación (sin intercambio de procesos completos) a disco, teniendo con una partición o un archivo en el sistema la posibilidad de añadir más áreas de intercambio según se necesite.
8. La memoria se gestiona como un recurso unificado para los programas de usuario y para la caché de disco, pudiendo utilizar la memoria libre para caché.
9. Posee librerías compartidas de carga dinámica (DLL's) y librerías estáticas.
10. Linux es casi totalmente compatible con POSIX (Portable Operating System Interface (for Unix)). POSIX es un estándar orientado a facilitar la creación de aplicaciones confiables y portables. La biblioteca para el manejo de hilos en POSIX es *pthread*, y permitirá efectuar múltiples tareas dentro de un mismo proceso gracias a los mecanismos de sincronización necesarios para conservar la consistencia de los recursos compartidos.
11. Soporta TCP/IP incluyendo ftp, telnet, NFS, etc..
12. Permite una gran escalabilidad; además en todo momento busca un compromiso directo con la seguridad asociada al sistema.

A continuación se muestra una tabla comparativa entre los sistemas operativos más comunes en la actualidad, que resultan de gran interés en este estudio de tecnologías:

Sistema Operativo	Conectividad	Confiabilidad	Escalabilidad	Multiusuario	Multiplataforma	Precio	Instalación	Interface	Seguridad
Windows 7	2	2	2	2	2	4	3	1	3
Mac OS X v10.6	1	2	2	1	1	3	2	1	1
Debian Linux	1	2	1	1	1	1	3	2	2
1. Excelente									
2. Bueno									
3. Regular									
4. Malo									

Tabla 1. Comparativa de Sistemas Operativos actuales

En esta tabla se pretende enfatizar las características de cada SO en relación al coste asociado y a las posibilidades reales que ofrece de cara al usuario. Todos los sistemas expuestos cumplen las especificaciones requeridas para poder soportar la plataforma desde la Word Wide Web, diferenciándose claramente en el precio de compra de cada uno de ellos. En términos de seguridad, la peor evaluación no corresponde al sistema menos seguro, sino al SO de mayor demanda puesto que es más factible que tenga ataques malintencionados.

En cuanto a la seguridad de la plataforma, aspecto que conlleva otro estudio asociado, cabe destacar que el servidor en el que se encuentra almacenado todo el software y las bases de datos pertenece a Linux (como se indicó anteriormente), puesto que se ha realizado un análisis exhaustivo de los servidores que soportan Windows y Linux y se ha concretado que este último presenta una mayor complejidad para los hackers, lo que le convierte en el SO más seguro.

En cuanto al entorno de usuario, se aconseja utilizar como sistema operativo la versión Debian de Linux (opcional). El Proyecto Debian es una asociación de personas que han desarrollado un SO libre. Este desarrollo de software libre hace que grupos de personas de muy diversa índole colaboren en su constante desarrollo para optimizar las características del mismo, independientemente de fechas concretas o entregas de lanzamiento. Investigadores de universidades, apasionados de la informática o simples desarrolladores de software que quieren devolver a la sociedad la posibilidad de crear sistemas accesibles y seguros para todos.

Para poder utilizar este sistema operativo hay que formalizar un documento denominado “Contrato Social”, por el cual se establecen los siguientes puntos: 1) Debian permanecerá 100% libre. 2) Contribuiremos a la comunidad de software libre. 3) No ocultaremos los problemas. 4) Nuestra prioridad son los usuarios y el software libre. 5) Trabajos que no siguen nuestros estándares de software libre.

Lenguajes de desarrollo

En cuanto al desarrollo de la plataforma, se va a implementar PHP (PHP Hypertext Pre-processor) como lenguaje de programación interpretado. Esta elección se ha determinado después de evaluar otra serie de lenguajes entre los que destaca Java.

PHP

PHP es un lenguaje que se puede desplegar en la mayoría de los servidores web y SO sin ningún tipo de coste. Tiene un reconocimiento internacional por encontrarse instalado en más de veinte millones de sitios web y en un millón de servidores.



PHP posee un gran parecido con los lenguajes más comunes de programación estructurada como C y Perl, por lo que resulta muy sencillo de interpretar y desarrollar para la mayoría de los programadores relacionados con estos lenguajes. Además su versatilidad permite que aunque su diseño esté orientado al desarrollo de portales web, también puede crear aplicaciones con una interfaz gráfica para el usuario, o ser usado desde la línea de comandos al igual que Perl o Python.

En cuanto a la relación directa que va a tener con las bases de datos, PHP permite una conexión con los diferentes tipos de servidores de BBDD existentes, tales como Oracle, MySQL, PostgreSQL, Microsoft SQL Server, ODBC, Solid, LDAP, filePro,, Sybase, DB2, Firebird y SQLite. Si nos basamos en los diferentes tipos de sistemas operativos, PHP puede ser ejecutado en Linux, Mac OS y Microsoft Windows con total normalidad.

En cuanto a las características asociadas a PHP, el siguiente listado hace alusión a las ventajas e inconvenientes que conlleva.

1. PHP es libre, lo cual representa una fácil accesibilidad para todos los usuarios.
2. Es un lenguaje multiplataforma, y orientado al desarrollo de aplicaciones web dinámicas con acceso a la información almacenada en una BBDD.
3. En cuanto a la conexión con gestores de bases de datos, destaca su conectividad con PostgreSQL y MySQL.
4. El código fuente desarrollado en PHP es invisible al navegador web y al usuario final, siendo una programación muy segura y confiable. El servidor se encargará de ejecutar el código y enviar el resultado HTML al navegador.
5. Permite aplicar técnicas de programación orientada a objetos.
6. Incluye una biblioteca amplia de funciones desde un primer momento.

7. Como inconveniente, al ser un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado.

Además de estas características, hay que destacar que debido a su diseño modular permite tener acceso a servidores IMAP, enviar correo con SMTP, acceder a SNMP para gestión de redes y equipos, generar dinámicamente gráficos y documentos pdf, analizar documentos XML y generar datos en WDDX (intercambio Web de Datos Distribuidos).

JAVA

Java es un lenguaje de programación, que utiliza desarrollo orientado objetos, y cuya sintaxis está basada en C++ para simplificar su aprendizaje, eliminando aspectos como el preprocesador de C, los typedefs y los punteros. El uso exclusivo del desarrollo orientado a objetos permite que no existan las funciones, salvo como métodos de acceso a una clase. De hecho las variables están incluidas dentro de las clases, para favorecer la encapsulación del código. Sus librerías de objetos predefinidas proporcionan también herramientas para las comunicaciones a través de la red.



El entorno de programación Java es un proyecto de la compañía Sun Microsystems cuyos principios son los siguientes:

1. Usar la metodología de la programación orientada a objetos.
2. Incluir por defecto el soporte necesario para poder trabajar en red, y así poder ejecutar código en sistemas remotos de forma segura.
3. Permitir la ejecución de java en un mismo programa o en múltiples sistemas operativos.

Las aplicaciones desarrolladas con Java podrán ejecutarse en entornos heterogéneos conectados a través de redes de comunicaciones, siendo eficientes y pudiendo crear dinámicamente en función de las necesidades existentes. Las características principales de este lenguaje son las siguientes:

1. Lenguaje simple, debido a su semejanza con lenguajes como C y C++ tal y como se ha comentado anteriormente.
2. Orientado a objetos, agrupando en ellos los datos y métodos utilizados.
3. Lenguaje distribuido, proporcionando una colección de clases para su uso en aplicaciones de red que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos, facilitando así la creación de aplicaciones distribuidas.
4. Interpretado y compilado a la vez, transformando su código fuente en una especie de código máquina, los bytecodes, similares a instrucciones de ensamblador. Además estos bytecodes se pueden ejecutar directamente sobre cualquier máquina a la cual se hayan portado el intérprete y el sistema de ejecución en tiempo real.
5. Lenguaje robusto y seguro. Se diseñó para crear software altamente fiable, ofreciendo numerosas comprobaciones en compilación y en tiempo de ejecución. Además incorpora barreras de seguridad en el lenguaje y en el sistema de ejecución en tiempo real para cubrir esta necesidad.
6. Es un lenguaje indiferente a la arquitectura, independiente de los diferentes entornos de red y de los diversos sistemas operativos. Para poder comportarse de esta manera existen los bytecodes, que utilizan un formato intermedio que no depende de la arquitectura propiamente dicha. Esto convierte a Java en un lenguaje portable, ofreciendo sus programas el mismo comportamiento en las diferentes plataformas.

Justificación de la propuesta

Después de analizar los lenguajes de programación existentes en la actualidad, y fundamentalmente PHP y Java, se ha determinado elaborar la plataforma en PHP. Java ofrece unas muy buenas metodologías de desarrollo para una aplicación de las características que requiere nuestra plataforma, pero sin embargo posee una excesiva lentitud y un abusivo consumo de memoria en comparación con PHP, lo que impide que los usuarios puedan tener un servicio óptimo, sobre todo teniendo en cuenta el módulo Sala Virtual, que precisa de recursos optimizados en tiempo real.

En cuanto a la seguridad que ofrecen ambos lenguajes para el desarrollo de la plataforma, Java ofrece unas prestaciones superiores que no obstante se suplen con notoriedad por el código PHP y la seguridad asociada al mismo.

Tal y como muestran las características asociadas a la descripción de PHP, la fuerte interoperabilidad con los gestores de bases de datos hacen dotar a la plataforma de una gran robustez y fiabilidad, sin limitar con ello el tiempo de respuesta de la misma.

Seguridad en el modelo cliente-servidor

Para garantizar un entorno seguro en la plataforma de telemedicina del consorcio RESATER se va a utilizar un protocolo de capa de conexión segura (**Secure Sockets Layer – SSL**), que va a permitir cifrar la conexión e incluso garantizar la autenticación mediante el uso de criptografía. La versión estandarizada por el IETF se denomina Transport Layer Security (TLS).

SSL implica una serie de fases como son la negociación entre las partes del algoritmo que utilizarán la comunicación, el intercambio de claves públicas y autenticación basada en certificados digitales, y por último el cifrado del tráfico basado en cifrado simétrico.

Concretamente, y desde un punto de vista objetivo, la tecnología SSL permite proteger las operaciones en línea ayudando a mejorar la confianza de la plataforma de telemedicina en base a estas tres premisas:

1. *Un certificado SSL permite habilitar un cifrado de información confidencial durante las operaciones en línea.*

Los servidores web y navegadores utilizan el protocolo SSL para facilitar un canal con un cifrado único que permite tener comunicaciones privadas a través de Internet. Para ello los cifrados SSL poseen una clave pública para cifrar la información, y una privada para descifrarla.

Cuando un navegador web accede a un dominio protegido, se establece un nivel de cifrado según el certificado SSL. Un cifrado robusto sería de 128 bits.

2. *Cada certificado SSL dota de una credencial única que identifica al propietario del certificado.*

Estas credenciales se utilizan en Internet emitiéndose de manera única para un determinado dominio y un servidor web por parte del proveedor del certificado SSL. Cuando un navegador establece una conexión con un servidor, éste le envía la información de identificación.

3. Una autoridad de certificación autentica la identidad del propietario que posee el certificado antes de emitirlo.

La confianza generada por una credencial depende directamente de la seguridad depositada en la autoridad de certificación que avala esa credencial.

A continuación se describe el protocolo a seguir por un navegador web cuando se conecta a la plataforma de telemedicina RESATER, protegida con Segure Sockets Layer.



Figura 1. Protocolo de establecimiento de conexión con SSL

Gestores de Bases de Datos

El siguiente mini-estudio va a tener relación con los gestores de bases de datos. Después de analizar todos los existentes en el mercado, se va a realizar una comparativa de los tres más utilizados: Oracle, MySQL y PostgreSQL.



PostgreSQL

PostgreSQL es un sistema de gestión de base de datos relacional orientado a objetos y libre, publicado bajo una licencia BSD, y dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista y/o apoyados por empresas concretas, denominándose PGDG (PostgreSQL Global Development Group).

Una característica muy importante de PostgreSQL es la alta concurrencia. Mediante un sistema de acceso concurrente multiversión (MVCC) se permite el acceso sin bloqueos a una tabla que a su vez está trabajando en un proceso. PostgreSQL también provee de una gran variedad de tipos nativos, ofreciendo soporte para bloques de direcciones estilo CIDR, direcciones MAC, arrays, texto de largo limitado, números de precisión arbitraria, direcciones IP (IPv4 e IPv6) y figuras geométricas (con una variedad de funciones asociadas).

Otras características que posee este gestor de BBDD son las claves ajenas (foreign keys), los disparadores (triggers), la herencia en tablas y el soporte para transacciones distribuidas. Esta última propiedad permite a PostgreSQL integrarse en un sistema distribuido formado por diferentes recursos (PostgreSQL, Oracle, mensajes IBM MQ JMS y ERP SAP) gestionado por un servidor de aplicaciones donde el éxito de la transacción global depende de lo que ocurra localmente.

MySQL

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario, que permite su uso bajo licencia GNU GPL pero que si se desea incorporar en productos privativos debe adquirir una licencia para este uso pues cuenta con el patrocinio de una empresa privada que posee el copyright de la mayor parte del código.

MySQL tiene una gran presencia en aplicaciones web como Drupal o phpBB, en plataformas Linux/Windows-Apache-MySQL-PHP/Perl/Python y en herramientas de seguimiento de errores. Su base de datos es muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación.

Las características más destacadas de este gestor de BBDD son el soporte ofrecido a multiplataforma, las vistas actualizables, los disparadores (triggers), el soporte para SSL, las transacciones con motores de almacenamiento InnoDB, BDB Y Cluster, la librería de datos embebida, el soporte completo para Unicode y los procedimientos almacenados. Además permite un sistema de contraseñas y privilegios seguro basado en la verificación del host, y utiliza tablas hash en memorias temporales.

Oracle

Oracle es un sistema de gestión de bases de datos objeto-relacional desarrollado por Oracle Corporation. Este gestor permite soporte multiplataforma, soporte de transacciones, estabilidad y escalabilidad, siendo un sistema fuerte y robusto ampliamente utilizado por las empresas en la actualidad. Además ha sido certificado para poder trabajar bajo GNU/Linux.

Sus características principales son las siguientes: entorno cliente/servidor, usuarios concurrentes, alto rendimiento en transacciones, disponibilidad controlada de los datos de las aplicaciones, gestión de la seguridad, compatibilidad y conectividad, adaptación a estándares de la industria como SQL-92, portabilidad y autogestión de la integridad de los datos.

El sistema Oracle es una herramienta muy potente que ofrece un gran rendimiento con operaciones simplificadas y eliminación del riesgo. Sin embargo el coste que conlleva lo hace un gestor de bases de datos que está siendo reemplazado por sistemas de similares características y que sin embargo no tienen coste alguno asociado como PostgreSQL y MySQL.

A continuación se expone una breve tabla comparativa entre los tres gestores de bases de datos en la cual se valora del 1 al 10 los diversos factores considerados.

Factores	PostgreSQL	MySQL	Oracle
Documentación	5	8	10
Instalación	10	5	7
Verificación	10	10	10
Creación del usuario	10	7	10
Primera consulta	10	10	10
Recursos	10	10	3

Tabla 2. Comparativa de Gestores de Datos

Después de todas las consultas realizadas, el sistema gestor de bases de datos de la plataforma de telemedicina del consorcio RESATER será **PostgreSQL**. Esta elección es consecuente con la premisa de utilizar software libre con la seguridad de que el sistema seleccionado permite una efectiva gestión de sus bases de datos. Las características que ofrece este gestor se adaptan a la solicitud requerida en el pliego de condiciones de la plataforma, y por lo tanto el desarrollo de esta herramienta será subyacente a los cuatro módulos de los que consta la plataforma para poder gestionar correctamente toda la información almacenada.

Diseño de la Arquitectura

La arquitectura de la plataforma de telemedicina, tal y como enuncia el Pliego de Condiciones del consorcio RESATER, estará formada por una estructura de capas. Este tipo de programación tiene como objetivo la independencia de la lógica de negocios con la lógica de diseño.

Las ventajas que supone este tipo de desarrollo son evidentes. En caso de tener que modificar cualquier aspecto relacionado con uno de los niveles, al encontrarse abstraído del resto, se puede operar en él sin modificar los niveles anexos. Este tipo de diseños, las arquitecturas multinivel o multicapa, sirven para definir en cada nivel una operación simple, contribuyendo a realizar un diseño altamente escalable puesto que se pueden ampliar con una mayor facilidad. El diseño más estandarizado es el que se realiza en tres capas, y por lo tanto será el aplicado en el desarrollo de la arquitectura de la plataforma de telemedicina.

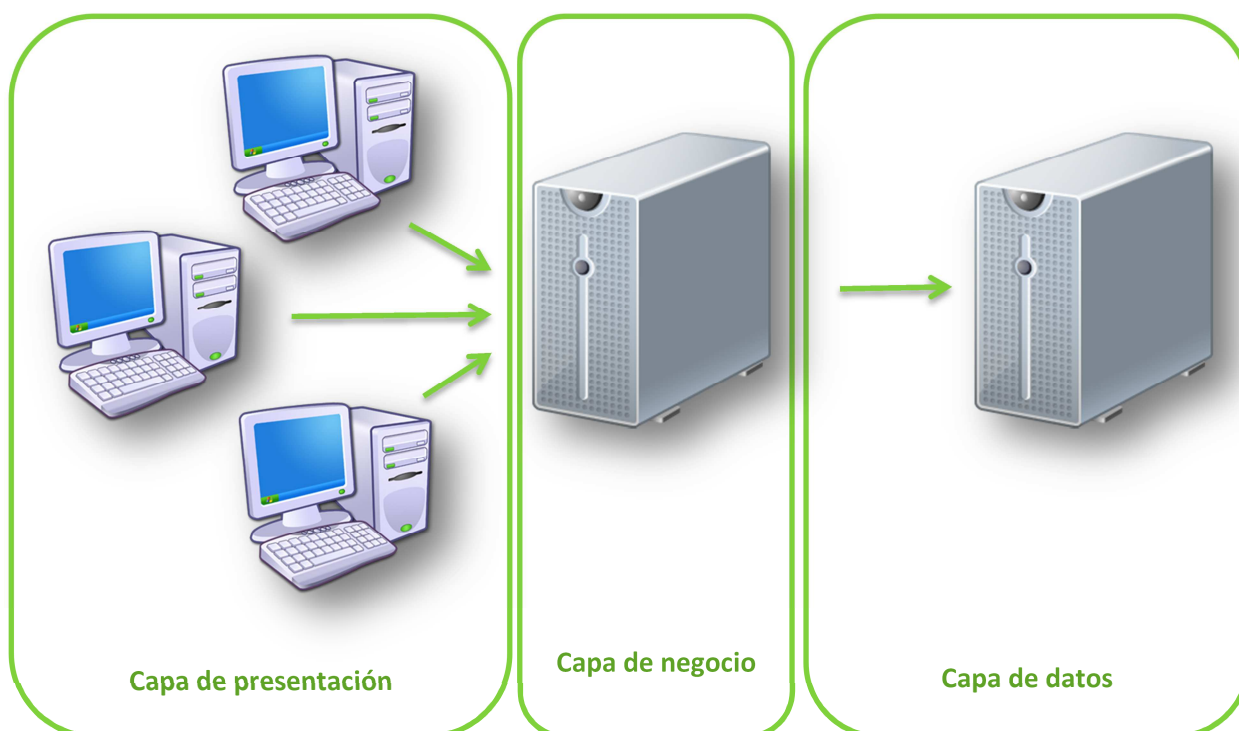


Figura 2. Arquitectura de la Plataforma de Telemedicina

1. **Capa de datos.** Esta capa es la de menor nivel y por lo tanto será la encargada de almacenar todos los datos y registros, permitiendo su acceso gracias al gestor de bases de datos (en este caso PostgreSQL). Esta capa estará en el servidor Linux sobre el que operará la plataforma, accediendo a ella desde la interfaz de usuario con los permisos correspondientes. Su eficaz desarrollo permitirá una óptima gestión de los datos de los profesionales, sus centros y sus ficheros asociados.
2. **Capa de negocio.** Esta capa es la encargada de llevar a cabo los procesos y las peticiones a realizar, puesto que contiene los programas que permiten todas estas acciones. La robusta programación de la capa de negocio con el lenguaje PHP tiene una doble función: interactuar con la interfaz de usuario de manera que el profesional quede abstraído de la capa de negocio, pero a su vez que se pueda beneficiar de la misma para poder acceder a las bases de datos correspondientes. Al igual que la capa de datos, el software correspondiente a esta capa estará almacenado en el servidor Linux de la plataforma.
3. **Capa de presentación.** Es la interfaz gráfica o de usuario con la que el profesional asociado a la plataforma RESATER podrá tener acceso a los diferentes módulos de la misma de una manera ágil y dinámica. Su sencillo diseño permitirá la fácil

interoperabilidad entre el conjunto de los usuarios con los ficheros almacenados en la plataforma, la disponibilidad de realizar videoconferencia y la gestión de archivos que se consideren de interés.

El desarrollo del software asociado a esta arquitectura se realizará con metodologías ágiles. Concretamente serán SCRUM para la gestión y organización, y eXtreme Programming (XP) para la ejecución, tal y como se cita en el Pliego de Condiciones.

Gracias a este tipo de metodologías se realizarán pequeñas tareas simples con sus correspondientes pruebas asociadas para conformar la estructura de la arquitectura de la plataforma de telemedicina. Este desarrollo permite una mejor distribución de las tareas a realizar por parte de SICBRAIN EUROPA SL resultando con ello una estructura coherente y flexible.

El resultado final implementará una interfaz de usuario (capa de presentación) en la cual existirá un gran compromiso entre los diferentes requisitos que un portal de estas características necesita.

1. Fiabilidad: Procesos de validación de usuario.
2. Funcionalidad: Capacidad de recuperación y servicios asociados a la plataforma.
3. Usabilidad: Capacidad de comprensión del portal web, estética dinámica y sencilla.
4. Mantenibilidad: Plataforma adaptable y extensible.
5. Eficiencia: Velocidad de generación de páginas y gráficos, rendimientos asociados.

La estructura de trabajo de la plataforma de telemedicina se desarrollará en siete fases claramente diferenciadas en las cuales, con la arquitectura multicapa anteriormente mencionada, se dará formato a los cuatro módulos pertenecientes a la plataforma, empezando en el relativo a la administración y gestión de los usuarios hasta finalizar con la sala virtual. En todos ellos habrá una primera fase de desarrollo de software en la capa de negocio que permitirá, por un lado, la implementación de la base de datos asociada a dicho módulo, y por otro la relación con la interfaz de usuario de la capa de presentación. Hay que tener en cuenta que la base de datos de la plataforma es conjunta en todos los módulos, desarrollando una

serie de autorizaciones que permitirán o denegarán el acceso a dichos gestores en función del usuario introducido en el sistema.

Conclusiones

Este Estudio 2 relativo a la arquitectura de la plataforma de telemedicina, tal y como se cita en la propuesta oferente de SICBRAIN EUROPA SL, pretende describir la estructura que va a tener el desarrollo de la plataforma desde sus inicios. Para ello se han descrito las tecnologías a utilizar en cada una de las capas, explicando la elección propuesta para su posterior implementación. Todo esto se realizará de conformidad con las especificaciones descritas en la documentación relativa a la plataforma del consorcio RESATER y a la documentación expuesta durante el proceso de licitación asociado.

En consonancia con el desarrollo técnico de los módulos, se realizará una eficaz base de datos gracias al gestor PostgreSQL que permita almacenar los datos de los usuarios que acceden a la plataforma, así como los relativos a los centros asociados, conversaciones mantenidas y documentación compartida. Todo esto *correrá* sobre un servidor Linux que almacenará el conjunto de la plataforma, estableciéndola en un entorno fiable y seguro.



Figura 3. Estructura de las tecnologías utilizadas

Para programar el conjunto de la plataforma se utilizará el lenguaje PHP puesto que su versatilidad y dinamismo se transfieren al usuario final al permitirle trabajar con un entorno ágil y robusto que posibilite una fácil interoperabilidad entre los profesionales asociados al consorcio RESATER. Este usuario final podrá tener acceso a la plataforma a través de cualquiera de los SO anteriormente mencionados en la tabla comparativa de los Sistemas Operativos actuales (tanto las versiones expuestas, como las inmediatamente inferiores),

teniendo todos ellos plenas garantías de respuesta para una plataforma de estas características.

La arquitectura multicapa de la plataforma permitirá realizar el diseño de la plataforma con una metodología de desarrollo independiente y dinámica. Este tipo de programación repercutirá favorablemente en la escalabilidad asociada a la misma.

Además, el conjunto de todas las tecnologías expuestas harán de la arquitectura de esta plataforma una sólida propuesta para el intercambio del conocimiento y la transferibilidad de modelos de práctica profesional y gestión de servicios en telemedicina.

Referencias

- [1] <http://www.linux-es.org/>
- [2] <http://www.debian.org/index.es.html>
- [3] <http://windows.microsoft.com/es-ES/windows-7?os=win7>
- [4] <http://www.apple.com/es/macosex/>
- [5] <http://www.php.net/>
- [6] <http://www.java.com/es/>
- [7] <http://www.verisign.es/ssl/ssl-information-center/>
- [8] <http://www.postgresql.org/>
- [9] <http://www.mysql.com/>
- [10] <http://www.oracle.com/es/index.html>